METHOD FOR DETERMINING COLOR SPACE OF AN IMAGE

Inventor: Rajá Bala

FIELD OF THE INVENTION

[0001] This invention relates generally to color management systems and methods, and more particularly to a method for automatically determining the color space of an input image

BACKGROUND OF THE INVENTION

[0002] Color image reproduction systems include an input device for obtaining a representation of an original image and an output device for generating a replica of the image. Input and output devices use signals that represent the colors in the image as coordinates in some device dependent color space. The term "color space" refers to an N-dimensional space in which each point corresponds to a particular color. Examples of device-dependent three-dimensional color space include RGB (red (R), green (G) and blue (B)) and CMY (cyan (C), magenta (M) and yellow (Y)). Frequently, it is advantageous to transform device dependent color images into device independent color images. Examples of device independent color spaces include luminance-chrominance color spaces such as CIELAB, YC_bC_r, and HSV (hue saturation value) space.

[0003] In order to provide the correct output, the output device must know in which color space the input image is defined. Similarly, an image processing module that takes as input a color image must know the color space in which the image is defined. Incorrect assumptions on the colorimetry of image data can result in unpredictable and usually undesirable output quality from a color management system. A rather extreme case of this is when the assumed class of color space, e.g, RGB vs. CMY vs. luminance-chrominance, is incorrect. Many file formats encode the color space class in the header. However, a robust

color management system must take into account cases where header information is incomplete, missing, or incorrect.

SUMMARY OF THE INVENTION

[0004] A method is described for automatically determining whether a 3-channel color image is in a device dependent color space such as RGB or a luminance-chrominance color space. The method may be used to determine whether an input image is in luminance-chrominance space (such as YCC, LAB or HSV) or in a device dependent color space, such as RGB or CMY, without reference to any header or auxiliary information in the image file. Color space differentiation can be used in isolation or in conjunction with other image processing or analysis algorithms. The method is based on the concept that in a luminance-chrominance representation, when the first channel (e.g., luminance) takes on very low or high values, there are constraints in the allowable values taken on by the other two chrominance channels. Namely, for colors with very high or very low luminance, the chrominance values are expected to be on or close to the neutral axis of the luminance-chrominance space. The neutral axis definition depends on the encoding of the color space. For example, in most 8-bit luminance-chrominance encodings, the chrominance values of a neutral color are both typically 128. Thus colors with very high or very low luminance values are expected to have chrominance values close to 128. These constraints do not hold for the RGB or CMY case. This concept is used to distinguish the two classes of color spaces.

[0005] A method for determining whether an input image is defined in accordance with a luminance-chrominance color space, according to the invention, includes receiving an input image in three dimensional color space, S1, S2 and S3, where S1 is an image value in the first dimension, S2 is an image value in the second dimension and S3 is an image value in the third dimension; extracting low and high extrema of S1 values in the image; obtaining deviation of S2 and S3 from a predefined neutral axis for all pixels in the image whose S1 value is either the low or the high extremum; testing a condition that certain of the deviations are within predetermined thresholds; and determining, if the condition holds true, that the

image is in a luminance-chrominance color space. If the condition is false, the image is in an RGB color space.

[0006] The entire input image need not be used; only a subset of the entire set of image pixels in the input image may be sampled. Extracting the low and high extrema of S1 values in the image may be accomplished by creating a histogram for S1 denoted by an array HIST, where HIST[i] is the number of image pixels whose S1 value is equal to i; defining the high extremum i_{high} to be the highest value of i for which HIST[i] is greater than a predetermined threshold K1; and defining the low extremum i_{low} to be the lowest value of i for which HIST[i] is greater than a predetermined threshold K2

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] Figure 1 is a flow chart of a method for determining color space of an image.

DETAILED DESCRIPTION OF THE EMBODIMENTS

[0008] Referring now to the drawing, Figure 1 is a flow chart of a method of determining the color space of an input image. Denote the three separations of an input image as S1, S2, and S3 (step 1). If the image is in luminance-chrominance, then very high or low values of S1, the luminance component, correspond to colors near white and black, respectively, both of which are neutral colors. In these cases, the other 2 channels, S2, and S3 should be similar in value, and close to the neutral value, e.g., 128 in an 8 bit representation. If the image is in RGB such a constraint does not exist. Very high or low values of S1 (i.e., red) in RGB color space could occur with any combination of the other 2 channels.

[0009] There are some cases where the aforementioned assumption may not hold. For example, suppose a saturated yellow is the lightest color in a luminance-chrominance representation. This results in both the S1 and yellow-blue dimensions taking on high values, and the red-green signal being in the vicinity of the neutral value. A similar situation occurs if a saturated blue is the darkest color in the image. However, the fact that there is a constraint on the red-green channel can be used to take care of both cases.

[0010] A specific implementation of the method to accomplish the classification is described below.

[0011] Referring again to Figure 1, create an array HIST which keeps a histogram of S1 (step 12). Define two arrays A1 and A2 of length 256 (adjusted according to input data precision). As the image is visited, A1 and A2 are built as follows:

A1[i] = max (S2, S3) over all pixels in the image for which S1 = I (step 14) and

A2[i] = min (S2, S3) over all pixels image for which S1 = I (step 16).

[0012] After the image has been traversed, define i_{low} as the lowest value of i for which HIST[i] is greater than some threshold, t_l (step 18). Define i_{high} as the highest values of i for which HIST[i] is greater than some threshold, t_2 (step 20). The thresholds t_l and t_2 are set to some small value in order to eliminate colors with very small pixel counts.

[0013] Next, define maximum deviations from neutral for the last two channels, occurring at S1 = i_{low} and S1= i_{high} (step 22):

D1 = abs(NEUTRAL – A1[i_{low}]);

D2 = abs(NEUTRAL – A2[i_{high}]);

D3 = abs(NEUTRAL – A2[i_{low}]); and

D4 = abs(NEUTRAL – A1[i_{high}]),

where abs() denotes absolute value, and NEUTRAL denotes the assumed neutral value in the given color space encoding. For example, in an 8-bit luminance-chrominance encoding, NEUTRAL is typically 128. Generally D1-D4 will take on small values in the case of luminance-chrominance images; while at least one of these will take on large values in the case of RGB images. To account for the blue and yellow cases described earlier for luminance-chrominance images, a larger range in the blue-yellow dimension must be allowed, while keeping a tighter constraint on the red-green dimension. Also, it must be noted that there are different flavors of luminance-chrominance spaces. In the YES and CIELAB spaces, the second and third dimensions represent red-green and yellow-blue variations respectively; while in YC_bC_r, the order and polarity are changed: the second and third

dimensions are now blue-yellow and red-green. A simple classification rule that takes all these factors into account is:

If (D1 < T1) AND (D2 < T2) (step 24) classify as luminance-chrominance (CIELAB case) (step 26)

else if (D3 < T3) AND (D4 < T4) (step 28) classify as luminance-chrominance (YC_bC_r case) (step 30)

else classify as RGB (step 32),

where T1 - T4 are predetermined thresholds chosen from analyzing a set of training images.

[0014] To increase the efficiency of the method, the image may be sub-sampled, with only a small subset of pixels being traversed. Also, some cases of RGB input can be detected before the full image is swept. For example, if pixels in the vicinity of [255,255,255], [255,0,0], [0,0,0] etc., are encountered, it is already known that the data is in RGB.

[0015] The method has been demonstrated to successfully distinguish between luminance-chrominance space and RGB for a large set of test images. Based on an analysis of the parameters D1-D4 for a set of training images, the following threshold values appear to produce good results for 8 bit image data: T1 = T3 = 25, T2 = T4 = 35.

[0016] One potential failure mode is the case where an RGB image, such as a test target, contains a small palette of colors chosen so that D1-D4 all take on small values, thus looking like a luminance-chrominance image to the algorithm. In such cases, there is insufficient context for the method to determine the color space. In fact, unless other contextual knowledge is available, this confusion may well occur even with visual inspection. Appropriate checks on i_{low} and i_{high}, as well as a count of the number of distinct image colors can help identify such pathological cases, and have the method report a low degree of confidence in classification.

[0017] In the case where there is no header information, misclassification of RGB vs. luminance-chrominance will result in gross errors in downstream imaging all the way to the final output. In the case where header information is available, the method serves as an extra

check to validate the color space, and can be combined with other image processing algorithms.

[0018] Another scenario where the method may be used is with documents comprising multiple image components assembled from different sources, and possibly in different color spaces. If the color space of each component is not individually labeled, this method can be used to extract this information, and attach the metadata to each component for correct processing by other imaging modules.

[0019] The invention has been described with reference to particular embodiments for convenience only. Modifications and alterations will occur to others upon reading and understanding this specification taken together with the drawings. The embodiments are but examples, and various alternatives, modifications, variations or improvements may be made by those skilled in the art from this teaching which are intended to be encompassed by the following claims.

[0020] The claims, as originally presented and as they may be amended, encompass variations, alternatives, modifications, improvements, equivalents, and substantial equivalents of the embodiments and teachings disclosed herein, including those that are presently unforeseen or unappreciated, and that, for example, may arise from applicants/patentees and others.